

Answers

Exercise 1

1. This is done using following command to generate the JSON file:

```
csv2bufr mappings create 001125 001126 001127 001128 004001 004002  
↪ 004003 004004 004005 005002 006002 007031 010004 012101 013009  
↪ --output mapping_1.json
```

The 6 digit numbers correspond to the following (identified by either the local BUFR tables or by looking up on the ECMWF website).

Parameter/variable	6 digit code (FXXYYY)
wigosIdentifierSeries	001125
wigosIssuerOfIdentifier	001126
wigosIssueNumber	001127
wigosLocalIdentifierCharacter	001128
year (UTC)	004001
month (UTC)	004002
day (UTC)	004003
hour (UTC)	004004
minute (UTC)	004005
latitude (coarse, 2dp)	005002
longitude (coarse, 2dp)	006002
heightOfBarometerAboveMeanSeaLevel (m)	007031
nonCoordinatePressure (Pa)	010004
airTemperature (K, 2dp)	012101
relativeHumidity (% , 1dp)	013009

2. The BUFR elements to delete are:

- “bufrHeaderCentre”
- “bufrHeaderSubCentre”

- “updateSequenceNumber”
- “dataSubCategory”
- “localTablesVersionNumber”
- “typicalSecond”
- “typicalDate”
- “typicalTime”
- “subsetNumber”

3. This is done using the `data transform` command as follows:

```
csv2bufr data transform --bufr-template mapping_1.json --output-dir .  
↪ ex_1.csv
```

This should create the following file: `WIGOS_0-20000-0-MYWSI001_20230203T120200.bufr4`

4. This can be done using the following command:

```
bufr_dump -p WIGOS_0-20000-0-MYWSI001_20230203T120200.bufr4 |grep -i  
↪ 'latitude\|longitude'
```

Exercise 2

We find the following BUFR sequences for the variables requested: * WIGOS station identifier (WIGOS identifier): 301150 * Date (Year, month, day): 301011 * Time (Hour, minute): 301012 * Location (Latitude/longitude (coarse accuracy)): 301023.

Using the BUFR codes for the remaining variables as in *exercise 1*, we can create a mappings template by the following command:

```
csv2bufr mappings create 301150 301011 301012 301023 007031 010004 012101  
↪ 013009 --output mapping_2.json
```

After deleting unnecessary BUFR elements in the mappings file like before, we convert `ex_2.csv` to BUFR using command:

```
csv2bufr data transform --bufr-template mapping_2.json --output-dir .  
↪ ex_2.csv
```

and of course can use BUFR Dump to inspect the contents of the output BUFR.

Exercise 3

1. As the data is the same, just some of the column names are styled differently, we can create the mapping file in the same way as *exercise 2*:

```
csv2bufr mappings create 301150 301011 301012 301023 007031 010004  
↪ 012101 013009 --output mapping_3.json
```

2. The elements to delete are the same as that in the previous exercises. As per the hint, we make the following changes to the mappings file:

```
"value": "data:wigosIdentifierSeries"  
→ "value": "data:wigos_identifier_series"
```

```
"value": "data:wigosIssuerOfIdentifier"  
→ "value": "data:wigos_issuer_of_identifier"
```

```
"value": "data:wigosIssueNumber"  
→ "value": "data:wigos_issue_number"
```

```
"value": "data:wigosLocalIdentifierCharacter"  
→ "value": "data:wigos_local_identifier_character"
```

```
"value": "data:latitude"  
→ "value": "data:lat"
```

```
"value": "data:longitude"  
→ "value": "data:lon"
```

```
"value": "data:heightOfBarometerAboveMeanSeaLevel"  
→ "value": "data:height_of_barometer_above_msl"
```

```
"value": "data:nonCoordinatePressure"  
→ "value": "data:non_coordinate_pressure"
```

```
"value": "data:airTemperature"  
→ "value": "data:air_temp"
```

```
"value": "data:relativeHumidity"
```

→ "value": "data:relative_humdity"

3. This is done as usual in the following way:

```
csv2bufr data transform --bufr-template mapping_3.json --output-dir .  
↪ ex_3.csv
```

4. Using the `bufr_dump` command as in *exercise 1* allows you to check that all the data is the same as the CSV.

Exercise 4

1. The sequence for the WIGOS station identifier is 301150 and the sequence for the synoptic data is 307080, thus we can create the mappings template as follows:

```
csv2bufr mappings create 301150 307080 --output mappings_4.json
```

2. Navigate to `answers/ex_4/synop_bufr.json` to see the correct mappings file for the example data. If you use your own CSV data, your mappings file will likely differ.

Exercise 5

1. The correct units are the following:

- `heightOfBarometerAboveMeanSeaLevel` should be in m, not cm.
- `nonCoordinatePressure` should be given in Pa, not hPa.
- `airTemperature` should be given in K, not °C.

2. The correct scale and offsets are:

Variable	Scale	Offset
<code>heightOfBarometerAboveMeanSeaLevel</code>	-2	0
<code>nonCoordinatePressure</code>	2	0
<code>airTemperature</code>	0	273.15

3. Your mappings file should contain the following:

```
{
  "eccodes_key": "#1#heightOfBarometerAboveMeanSeaLevel",
  "value": "data:height_of_barometer_above_msl",
  "valid_min": "const:-400.0",
  "valid_max": "const:12707.0",
  "scale": "const:-2",
  "offset": "const:0"
},
{
  "eccodes_key": "#1#nonCoordinatePressure",
  "value": "data:non_coordinate_pressure",
  "valid_min": "const:0",
  "valid_max": "const:163820",
  "scale": "const:2",
  "offset": "const:0"
},
{
  "eccodes_key": "#1#airTemperature",
  "value": "data:air_temp",
  "valid_min": "const:0.0",
  "valid_max": "const:655.34",
  "scale": "const:0",
  "offset": "const:273.15"
},
}
```

4. This is done as usual in the following way:

```
csv2bufr data transform --bufr-template mapping_5.json --output-dir .
↪ ex_5.csv
```

Question 6

1. The `relative_humidity` variable cannot take a value of -5% because it is impossible for relative humidity to be less than 0.
2. The `relative_humidity` variable is at minimum 0, and the maximum value is subjective, for example 150% .
3. The `relative_humidity` element should be adjusted to look like the following:

```
{
  "eccodes_key": "#1#relativeHumidity",
```

```
"value": "data:relative_humidity",  
"valid_min": "const:0",  
"valid_max": "const:150",  
"scale": "const:0",  
"offset": "const:0"  
}
```

4. You should repeat the process above to the variables of your choice, keeping the units of the input data in mind. For example, as `air_temp` is in °C you may set `"valid_min": "const:-20"` and `"valid_max": "const:40"`.
5. This is done as usual in the following way:

```
csv2bufr data transform --bufr-template mapping_6.json --output-dir .  
↪ ex_6.csv
```

```
bufr_dump -p WIGOS_0-20000-0-MYWSI001_20230203T120200.bufr4 |grep -i  
↪ 'relativeHumidity'
```

which should output:

```
relativeHumidity=MISSING
```